

Projects  
Development of Thermal Neutral Formats - Phase 2  
Network-model Results Format  
ESA Contract No. 10596/93/NL/FG Rider No. 1 and No. 2

# Space-domain Integrated Resources

**SIR-001-IR**

**Version 2**

**Release 2.1 (28 March 1997)**



**European Space Agency**



**French Space Agency**



**Fokker Space**



**Association GOSET**



---

# Table of Contents

---

Abstract.....	ii
Document Status Sheet.....	iii
Table of Contents.....	v
Introduction .....	ix
1 - Scope.....	1
1.1      Product versioning .....	1
1.2      Parameterized functions.....	1
1.3      Network model .....	1
1.4      Property class.....	1
1.5      Analysis, test or operation campaign.....	1
1.6      Measure.....	2
1.7      Space mission .....	2
1.8      Boolean surfaces .....	2
1.9      Thermal radiative meshing .....	2
1.10     Colour assignment .....	2
1.11     Kinematics .....	2
2 - Normative References.....	3
3 - Definitions and Abbreviations.....	5
3.1      Terms Defined in ISO 10303-1 .....	5
3.2      Terms Defined in this Part.....	5
3.3      Abbreviations.....	6
4 - SIR_product_version_schema .....	7
4.1      Introduction.....	7
4.2      Fundamental concepts and assumptions .....	7
4.3      SIR_product_version_schema entity definitions .....	7
5 - SIR_parameterized_function_schema .....	9
5.1      Introduction.....	9
5.2      Fundamental concepts and assumptions .....	9
5.3      SIR_parameterized_function_schema type definitions.....	10
5.4      SIR_parameterized_function_schema entity definitions .....	11
5.5      SIR_parameterized_function_schema function definitions .....	14
6 - SIR_network_model_schema .....	17
6.1      Introduction.....	17
6.2      Fundamental concepts and assumptions .....	17

---

6.3	SIR_network_model_schema type definitions .....	18
6.4	SIR_network_model entity definitions .....	18
7 - SIR_property_schema.....		23
7.1	Introduction .....	23
7.2	Fundamental concepts and assumptions .....	23
7.3	SIR_property_schema type definitions.....	23
7.4	SIR_property_schema entity definitions .....	24
7.5	SIR_property_schema function definitions .....	29
8 - SIR_ato_campaign_schema.....		31
8.1	Introduction .....	32
8.2	Fundamental concepts and assumptions .....	32
8.3	SIR_ato_campaign_schema type definitions.....	33
8.4	SIR_ato_campaign_schema entity definitions .....	34
9 - SIR_measure_schema.....		43
9.1	Introduction .....	43
9.2	Fundamental concepts and assumptions .....	43
9.3	SIR_measure_schema entity definitions.....	44
10 - SIR_space_mission_schema.....		47
10.1	Introduction .....	48
10.2	Fundamental concepts and assumptions .....	48
10.3	space_mission_schema type definitions .....	48
10.4	SIR_space_mission_schema entity definitions.....	50
10.5	SIR_space_mission_schema function definitions .....	57
11 - SIR_boolean_surfaces_schema .....		59
11.1	Introduction .....	59
11.2	Fundamental concepts and assumptions .....	59
11.3	SIR_boolean_surfaces_schema type definitions .....	59
11.4	SIR_boolean_surfaces_schema entity definitions .....	60
12 - SIR_thermal_radiative_meshing_schema .....		63
12.1	Introduction .....	63
12.2	Fundamental concepts and assumptions .....	63
12.3	SIR_thermal_radiative_meshing_schema type definitions .....	63
12.4	SIR_thermal_radiative_meshing_schema entity definitions .....	64
12.5	SIR_thermal_radiative_meshing_schema function definitions .....	70
13 - SIR_colour_assignment_schema.....		73
13.1	Introduction .....	73
13.2	Fundamental concepts and assumptions .....	73
13.3	SIR_colour_assignment_schema entity definitions.....	73

---

---

<b>14 - SIR_kinematics_schema .....</b>	<b>75</b>
14.1 <b>Introduction.....</b>	<b>75</b>
14.2 <b>Fundamental concepts and assumptions .....</b>	<b>75</b>
14.3 <b>SIR_kinematics_schema entity definitions .....</b>	<b>76</b>
<b>Annex A - Short names of entities (normative).....</b>	<b>79</b>
<b>Index.....</b>	<b>81</b>

## 6.3 SIR\_network\_model\_schema type definitions

### 6.3.1 SIR\_model\_constituent

The SIR\_model\_constituent type provides a mechanism to identify the kinds of elements a SIR\_model may contain.

#### EXPRESS specification:

```
*)  
TYPE SIR_model_constituent = SELECT(  
    SIR_node,  
    SIR_node_relationship,  
    SIR_submodel_usage  
);  
END_TYPE;  
(*
```

### 6.3.2 SIR\_node\_or\_usage

The SIR\_node\_or\_usage type provides a mechanism to identify the kinds of elements a SIR\_node\_relationship may relate.

#### EXPRESS specification:

```
*)  
TYPE SIR_node_or_usage = SELECT(  
    SIR_node,  
    SIR_node_usage  
);  
END_TYPE;  
(*)
```

### 6.3.3 SIR\_network\_component

The SIR\_network\_component type provides a mechanism to identify the kinds of elements that may be collected in a SIR\_component\_sequence.

#### EXPRESS specification:

```
*)  
TYPE SIR_network_component = SELECT(  
    SIR_model,  
    SIR_node,  
    SIR_node_usage,  
    SIR_node_relationship,  
    SIR_node_relationship_usage,  
    SIR_submodel_usage  
);  
END_TYPE;  
(*)
```

## 6.4 SIR\_network\_model entity definitions

### 6.4.1 SIR\_model

A SIR\_model is a kind of representation that collects SIR\_model\_constituents in order to form a representation of a space-domain product suitable for analysis, test or operation activities.

A SIR\_model has an identifier and may have a version identifier.

#### EXPRESS specification:

```
*)  
ENTITY SIR_model  
SUBTYPE OF (representation);
```

## 6 - SIR\_network\_model\_schema

The following EXPRESS declaration begins the SIR\_network\_model\_schema and identifies the necessary external references.

EXPRESS specification:

```
*)  
SCHEMA SIR_network_model_schema;  
  
REFERENCE FROM representation_schema (  
    representation,  
    representation_item,  
    mapped_item  
);  
  
REFERENCE FROM qualified_measure_schema (  
    type_qualifier  
);  
  
REFERENCE FROM support_resource_schema (  
    label,  
    text,  
    identifier  
);  
(*
```

**NOTE**

- 1) The schemas referenced above can be found in the following parts:

representation_schema	ISO 10303-43
support_resource_schema	ISO 10303-41
qualified_measure_schema	ISO 10303-45

### 6.1 Introduction

The subject of the SIR\_network\_model\_schema is the definition of representations of space-domain products in the form of network models containing nodes and relationships between nodes.

### 6.2 Fundamental concepts and assumptions

For the test or analysis purposes, space-domain products are usually represented as a collection of nodes and of relationships between these nodes. These nodes and relationships are used to identify the places in the products where quantities are predicted, assigned, observed or computed.

In the context of this schema, such a representation is called network model.

This schema specifies the resources to describe network models containing:

- nodes,
- relationships between two or more nodes (e.g. couplings),
- occurrences of submodels.

When a occurrence of a network model is created as a submodel of another one, this creates implicit nodes or relationships as occurrences of the nodes and relationships of the submodel. The SIR\_network\_model\_schema provides resources enabling to distinguish these implicit nodes and relationships and to assign to them properties.

**Formal propositions:**

WR1: the lower\_bound\_expressions attribute shall define valid bounds for the parameters of the function;

WR2: the upper\_bound\_expressions attribute shall define valid bounds for the parameters of the function;

**Informal propositions:**

IP1: the elements of polynom\_expression shall only be of types:

- simple\_n\_expression,
- plus\_expression,
- mult\_expression, or
- power\_expression

IP2: the second operand of any instance of power\_expression shall be of type int\_literal.

#### 5.4.4 SIR\_tabular\_function

A SIR\_tabular\_function is a kind of SIR\_parameterized\_function where the functions is specified by tables of values and a method to interpolate them.

The ordering of the result values shall be such that the index for the highest number parameter\_properties changes fastest.

EXAMPLE 1 – let be a SIR\_tabular\_function  $T(a_i, b_j)$  depending on two variables:  $a$  with index  $i$  and  $b$  with index  $j$ . For  $a$  3 values are specified and for  $b$  5 values are specified.

Then parameter\_values has the following format:  $((a_1, a_2, a_3), (b_1, b_2, b_3, b_4, b_5))$ ,

and result\_values shall be specified as:  $(T(a_1, b_1), T(a_1, b_2), T(a_1, b_3), T(a_1, b_4), T(a_1, b_5),$

$T(a_2, b_1), T(a_2, b_2), T(a_2, b_3), T(a_2, b_4), T(a_2, b_5), T(a_3, b_1), T(a_3, b_2), T(a_3, b_3), T(a_3, b_4), T(a_3, b_5))$

**EXPRESS specification:**

```
*)
ENTITY SIR_tabular_function
SUBTYPE OF (SIR_parameterized_function);
  parameter_values: LIST[1:?] OF listAscendingRealiterals;
  result_values: listRealiterals;
  interpolation_type: tabular_interpolation;
  interpolation_degree: tabular_interpolation_degree;
WHERE
  WR1: ((interpolation_type=polynomial) AND (EXISTS(interpolation_degree))) OR
  (interpolation_type=linear_logarithmic);
  WR2: SIZEOF(SELF\SIR_parameterized_function.parameters) = SIZEOF(parameter_values);
  WR3: consistent_size(parameter_values, result_values);
END_ENTITY;
(*
```

**Attribute definitions:**

**parameter\_values:** specifies the lists of ascending values associated with each parameter of the function;

**result\_values:** specifies the list of values defining the result of the function;

**interpolation\_type:** specifies the kind of interpolation that shall be applied;

**interpolation\_degree:** specifies the degree for the interpolation when the interpolation is polynomial.

**Formal propositions:**

WR1: if the interpolation is polynomial then the degree shall be specified;

WR2: there shall be as much list of values in parameter\_values as there are parameters;

WR3: the number of values in result\_values shall be consistent with the numbers of values specified for the parameters.

**WR2:** either the variable has no particular semantics or its semantics of the entity is defined by an instance of **SIR\_property\_usage** that refers to a **SIR\_property\_quantitative**.

NOTE - **SIR\_property\_quantitative** is defined in the schema **SIR\_property\_schema**.

#### 5.4.2 SIR\_parameterized\_function

A **SIR\_parameterized\_function** is the specification of the common characteristics of **SIR\_polynomial\_function** and **SIR\_tabular\_function**.

These common characteristics are the specification of the name of the function, possibly its mathematical formula, additional descriptive information and the identification as **SIR\_variables** of the input parameters and of the result. Only instances of specialization of **SIR\_parameterized\_function** are permitted.

##### EXPRESS specification:

```
*)  
ENTITY SIR_parameterized_function  
ABSTRACT SUPERTYPE OF (ONEOF( SIR_polynomial_function, SIR_tabular_function));  
    name: label;  
    description: text;  
    formula: OPTIONAL mathematical_string;  
    parameters: LIST[1:?] OF UNIQUE SIR_variable;  
    result: SIR_variable;  
END_ENTITY;  
(*
```

##### Attribute definitions:

**name:** specifies the name under which the function is referred to by users;

**description:** specifies some textual description of the function and of its intended usage;

**formula:** specifies the formula of the function. The formula attribute need not be specified for a particular **SIR\_parameterized\_function**;

**parameters:** specifies the variables of the function;

**result:** specifies the resulting quantity as a variable.

#### 5.4.3 SIR\_polynomial\_function

A **SIR\_polynomial\_function** is a kind of **SIR\_parameterized\_function** where the function is expressed by a polynom of one or more variables.

The domain of validity of each input parameter may be specified. When not specified, the lower bound for an input parameter is: - infinite. When not specified, the upper bound for an input parameter is: + infinite.

##### EXPRESS specification:

```
*)  
ENTITY SIR_polynomial_function  
SUBTYPE OF (SIR_parameterized_function);  
    lower_bound_expressions: SET[0:?] OF comparison_less_equal;  
    upper_bound_expressions: SET[0:?] OF comparison_greater_equal;  
    polynom_expression: numeric_expression;  
WHERE  
    WR1:valid_bounded_variables(lower_bound_expressions,  
SELF\SIR_parameterized_function.parameters);  
    WR2:valid_bounded_variables(upper_bound_expressions,  
SELF\SIR_parameterized_function.parameters);  
END_ENTITY;  
(*)
```

##### Attribute definitions:

**lower\_bound\_expressions:** specifies the minimum values of the parameters;

**upper\_bound\_expressions:** specifies the maximum values of the parameters;

**polynom\_expression:** specifies, as a formal expression, the polynom.

---

```

WHERE
  WR1: SELF >= 1;
END_TYPE;
(*

```

Formal proposition:

WR1: the degree of interpolation shall be greater than 0.

### 5.3.4 list\_real\_literals

The **list\_real\_literals** type provides a mechanism to refer to list of real values associated with the same quantity.

Express specification:

```

*)
TYPE list_real_literals = LIST[1:?] OF REAL;
END_TYPE;
(*

```

### 5.3.5 listAscendingReal\_literals

The **listAscendingReal\_literals** type provides a mechanism to refer to a list of real values defining the successive values of a parameter of a **SIR\_tabular\_function**.

Express specification:

```

*)
TYPE listAscendingReal_literals = list_real_literals;
WHERE
  WR1: ascending_list(SELF);
END_TYPE;
(*

```

Formal proposition:

WR1: the list of values shall be in a monotonic ascending order.

## 5.4 SIR\_parameterized\_function\_schema entity definitions

### 5.4.1 SIR\_variable

A **SIR\_variable** is the specification of a variable and its association with a **SIR\_property\_usage** that defines its semantics.

Note 3 ~ **SIR\_property\_usage** is defined in the schema **SIR\_ato\_campaign\_schema**.

EXPRESS specification:

```

*)
ENTITY SIR_variable
SUBTYPE OF (numeric_variable);
WHERE
  WR1: SIZEOF(SELF\generic_variable.interpretation)=1;
  WR2: (SIZEOF(SELF\generic_variable.interpretation)=0) OR
    ('SIR_ATO_CAMPAIGN_SCHEMA.SIR_PROPERTY_USAGE' IN
    TYPEOF(SELF\generic_variable.interpretation[1].semantics)) AND
    ('SIR_PROPERTY_SCHEMA.SIR_PROPERTY_QUANTITATIVE' IN
    TYPEOF(SELF\generic_variable.interpretation[1].semantics.property)));
END_ENTITY;
(*

```

Formal propositions:

WR1: there shall be exactly one element in the inherited attribute **SELF\generic\_variable.interpretation**, i.e. the entity shall be associated with exactly one instance defining its semantics;

Therefore, the definition of a parameterized function implies to specify:

- the variables (input and output),
- possibly their domain of validity, and
- the expression that enable to compute the result from the data of the parameters.

The present schema deals with functions such that:

- the input parameters and the result are NUMBERS,
- the function itself is either polynomial or tabular.

Note 2 – the type NUMBER is defined in ISO 10303-11

In the context of this schema:

- A variable is defined with two aspects: the first enables to identify and to refer to the variable in the function; the second enables to assign a meaning to the variable. This meaning is obtained through a reference to the usage of a property class in the context of an *ato\_case*;
- a polynomial function is characterized by the formal expression of the polynom and, possibly, by the specification of validity limits for the input parameters;
- a tabular function is defined by the a set of n-uples specifying the values of the input parameters and a corresponding set of values specifying the value of the result. In addition, complementary information enables to interpolate the function between the n-uples of parameter values.

## 5.3 SIR\_parameterized\_function\_schema type definitions

### 5.3.1 less\_or\_greater

The **less\_or\_greater** type provides a mechanism to refer either to a **comparison\_less\_equal** entity or to a **comparison\_greater\_equal** entity.

EXPRESS specification:

```
*)  
TYPE less_or_greater = SELECT(  
    comparison_less_equal,  
    comparison_greater_equal);  
END_TYPE;  
(*)
```

### 5.3.2 tabular\_interpolation

The **tabular\_interpolation** ENUMERATION type provides a mechanism to specify what kind of interpolation is considered for a given **SIR\_tabular\_function**.

EXPRESS specification:

```
*)  
TYPE tabular_interpolation = ENUMERATION OF (  
    polynomial,  
    linear_logarithmic);  
END_TYPE;  
(*)
```

### 5.3.3 tabular\_interpolation\_degree

The **tabular\_interpolation\_degree** type specifies the allowed values for the degree of a polynomial interpolation of n **SIR\_tabular\_function**.

EXPRESS specification:

```
*)  
TYPE tabular_interpolation_degree = INTEGER;
```

## 5 - SIR\_parameterized\_function\_schema

The following EXPRESS declaration begins the **SIR\_parameterized\_function\_schema** and identifies the necessary external references.

### EXPRESS specification:

```
*)  
SCHEMA SIR_parameterized_function_schema;  
  
REFERENCE FROM ISO_13584_generic_expressions_schema(  
    variable_semantics,  
    environment,  
    generic_variable,  
    binary_gen_expression  
);  
  
REFERENCE FROM ISO_13584_expressions_schema(  
    int_literal,  
    real_literal,  
    numeric_variable,  
    numeric_expression,  
    plus_expression,  
    minus_expression,  
    mult_expression,  
    power_expression,  
    comparison_less_equal,  
    comparison_greater_equal  
);  
  
REFERENCE FROM ISO13584_IEC1360_dictionary_schema(  
    mathematical_string  
);  
  
REFERENCE FROM support_resource_schema(  
    label,  
    text  
);
```

(\*  
NOTE

- 1) The schemas referenced above can be found in the following parts of ISO 13584 and ISO 10303:

ISO_13584_generic_expressions_schema	ISO 13584-20
ISO_13584_expressions_schema	ISO 13584-20
ISO13584_IEC1360_dictionary_schema	ISO 13584-42
support_resource_schema	ISO 10303-41

### 5.1 Introduction

The subject of the **SIR\_parameterized\_function\_schema** is the definition of resources enabling to describe numerical functions of one or more parameters as instances of EXPRESS entities.

### 5.2 Fundamental concepts and assumptions

The present schema deals with the concept of parameterized function. This concept is usually summarized by a symbolic expression  $y=F(x)$ , where  $x$  represents one or more measurable parameters,  $F$  the function itself and  $y$  the quantity which results from the computation of  $F$  with the data of  $x$ .

- 3.2.5 **property** - a quality or attribute belonging and especially peculiar to an object, common to all members of a class.
- 3.2.6 **qualifier** - a word or word group that limits or modifies the meaning of another word or word group [Webster].
- 3.2.7 **root-model** – The highest level model in a hierarchical model/submodel tree. The root-model comprises directly or indirectly all submodels.
- 3.2.8 **state** - mode or condition of being [Webster].
- 3.2.9 **submodel** – a next lower level model in a hierarchical model/submodel tree. A submodel may contain other (yet lower level) submodels. The model/submodel tree forms an acyclic graph.
- 3.2.10 **submodel occurrence** – a reference to a model at a level lower than root-model level in a hierarchical model/submodel tree. The submodel occurrence has some attributes in addition to the model it references: e.g. its usage name and – for geometric models – its location and orientation with respect to its supermodel.
- 3.2.11 **supermodel** – a next higher level model in a hierarchical model/submodel tree. A supermodel contains one or more submodels. The model/submodel tree forms an acyclic graph.
- 3.2.12 **tensor** - a generalization of the concept of vector that consists of a set of components usually having multiple rows of indices that are functions of the coordinate system and have invariant properties under transformation of the coordinate system. [Webster]
- 3.2.13 **unit** - a determinate quantity (as of length, time, heat, value, or housing) adopted as a standard of measurement [Webster]

### 3.3 Abbreviations

AP	Application Protocol
ato	Analysis, Test or Operation
CNES	Centre National d'Etudes Spatiales – French Space Agency
ESA	European Space Agency
ESTEC	ESA - European Space Technology Centre
ISO	International Organization for Standardization
PLib	Parts Library (ISO 13584)
SET	<i>Standard d'Echange et de Transfert</i> (French Standard AFNOR NF Z68-300)
SET-ATS	SET Protocol d'Application Thermique Spatiale
SI	<i>Système International des Unités</i> - International System of Units
STEP	Standard for the Exchange of Product Model Data (ISO 10303)
STEP-TAS	STEP-based Thermal Analysis for Space
TBD	To be defined

## 3 - Definitions and Abbreviations

---

### 3.1 Terms Defined in ISO 10303-1

- 3.1.1 **application** – a group of one or more processes creating or using product data.
- 3.1.2 **application protocol** – a document that specifies an application interpreted model obtained with specializing integrated resources in order to satisfy the scope and information requirements for a specific application.
- 3.1.3 **data** – a representation of information in a formal manner suitable for communication, interpretation, or processing by human beings or computers.
- 3.1.4 **data exchange** – the storing, accessing, transferring, and archiving of data.
- 3.1.5 **data specification language** – a set of rules for defining data and their relationships suitable for communication, interpretation, or processing by computers.
- 3.1.6 **exchange structure** – a computer-interpretable format used for storing, accessing, transferring, and archiving data.
- 3.1.7 **implementation method** – a part of ISO 10303 that specifies a technique used by computer systems to exchange product data that is described using the EXPRESS data specification language [STEP-11].
- 3.1.8 **information** – facts, concepts, or instructions.
- 3.1.9 **information model** – a formal model of a bounded set of facts, concepts or instructions to meet a specified requirement.
- 3.1.10 **integrated resource** – a part of this International Standard (i.e. ISO 10303) that defines a group of resource constructs used as the basis for product data.
- 3.1.11 **product** – a thing or substance produced by a natural or artificial process.
- 3.1.12 **product data** – a representation of information about a product in a formal manner suitable for communication, interpretation, or processing by human beings or by computers.
- 3.1.13 **product information** – facts, concepts, or instructions about a product.
- 3.1.14 **product information model** – an information model which provides an abstract description of facts, concepts and instructions about a product.
- 3.1.15 **resource construct** – a collection of EXPRESS language entities, types, functions, rules and references that together define a valid description of an aspect of product data.
- 3.1.16 **structure** – a set of interrelated parts of any complex thing, and the relationships between them.

### 3.2 Terms Defined in this Part

- 3.2.1 **ato\_case** - the gathering of a scenario of analysis or test, of the initial and boundary conditions and of the object to be analysed or tested.
- 3.2.2 **class** – a named category of items which share common characteristics and behaviour.
- 3.2.3 **exchange dataset** – a coherent and valid set of instances of entities conforming to the AIM schema defined in this AP. The dataset is instantiated in the form of a [STEP-21] physical file, a working form or a shared database. An exchange dataset can be accessed through an SDAI programming interface, as defined in [STEP-22].
- 3.2.4 **model** – a representation in software or hardware of one or more properties of a product and possibly its operational environment for the purpose of design, analysis or verification. By its nature a model is an idealised representation of the product.

The following are outside the scope of this part:

- description of a run and of a sequence of runs;
- specialization of the concept of scan to depict scans derived from other scans.

## 1.6 Measure

The following are within the scope of the SIR\_measure\_schema:

- definition of a conversion\_based\_unit with offset;
- association of a value with a property class;
- specification of a couple (value, property class) as an item of representation.

## 1.7 Space mission

The following are within the scope of the SIR\_space\_mission\_schema:

- identification of an ato case of a space mission;
- identification of a phase in an space ato case, bounded by two events;
- identification of events or contexts occurring during a space mission;
- description of a discretised orbit arc;
- description of a keplerian orbit arc;
- description of the positioning and orientation of a space product;
- description of the rotation motion of a space product.

## 1.8 Boolean surfaces

The following are within the scope of the SIR\_boolean\_surfaces\_schema:

- definition of an entity as the result of a boolean operation applied to a face;
- definition of a replica of such an entity.

## 1.9 Thermal radiative meshing

The following are within the scope of the SIR\_thermal\_radiative\_meshing\_schema:

- definition of a thermal radiative face;
- specification of the meshing of a thermal radiative face.

## 1.10 Colour assignment

The following are within the scope of the SIR\_colour\_assignment\_schema:

- assignment of a colour to some product data that may be graphically displayed.

## 1.11 Kinematics

The following are within the scope of the SIR\_kinematics\_schema:

- specification of a revolute pair as the result of the orientation of a kinematic link;
- specification of the articulation of a kinematic joint as a space mission aspect;
- specification of a kinematic link representing a component of a space product.

# 1 - Scope

---

This part specifies the resource constructs needed to describe the specific aspects of the spacecraft products, of their design and of their simulation or tests.

## 1.1 Product versioning

The following are within the scope of the SIR\_product\_version\_schema:

- identification of a version of a product;

## 1.2 Parameterized functions

The following are within the scope of the SIR\_parameterized\_function\_schema:

- specification of a function of one or more variables;
- specification of a variable;
- specification of a polynomial function of one or more parameters, including the specification of the domain of applicability of each parameter;
- specification of a tabular function of one or more parameters;
- specification of a periodic tabular function of one parameter.

## 1.3 Network model

The following are within the scope of the SIR\_network\_model\_schema:

- specification of a model based on a network of nodes;
- specification of the constituents of such a model;
- specification of the usage of a submodel in a model;
- identification of constituents of a model when it is used as a submodel of another model;
- specification of a sequence of constituents of a model.

## 1.4 Property class

The following are within the scope of the SIR\_property\_schema:

- specification of a class of property;
- specification of a scalar class of property;
- specification of specializations of a scalar property, the values of which are either chosen in a list of strings, or a list of parameterized functions or are numeric;
- specification of a class of property expressed as a tensor of scalar properties;
- characterization of the meaning of a property as a list of qualifiers.

## 1.5 Analysis, test or operation campaign

The following are within the scope of the SIR\_ato\_campaign\_schema:

- specification of a campaign of test or analysis;
- specification of a test or analysis case in a campaign;
- specification of a phase in a case;
- specification of the recording of the values of a property;
- specification of the usage of a class of a property in the context of a case and with a particular role;
- specification of the assignment of a unit to a class of property in the context of a case.